**UCS1712 – GRAPHICS AND MULTIMEDIA LAB**

**Lab Exercise 9: 3-Dimensional Projections in C++ using OpenGL**

**CODE:**

#include<gl/glut.h>

#include<iostream>

using namespace std;

int alpha = 0, theta = 0;

bool flag = true;

void init() {

glClearColor(1.0, 1.0, 1.0, 1.0);

glEnable(GL\_DEPTH\_TEST);

}

void keyPress(int key, int x, int y) {

switch (key) {

case GLUT\_KEY\_RIGHT: alpha++;

break;

case GLUT\_KEY\_LEFT:

alpha--;

break;

case GLUT\_KEY\_UP:

theta++;

break;

case GLUT\_KEY\_DOWN:

theta--;

break;

case GLUT\_KEY\_HOME: flag = !flag;

break;

}

glutPostRedisplay();

}

void display() {

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

if (flag) glOrtho(-500, 500, -500, 500, -500, 500);

else gluPerspective(100, 1, 0.1, 1000);

glMatrixMode(GL\_MODELVIEW);

glLoadIdentity();

gluLookAt(0, 0, 300, 0, 0, 0, 0, 1, 0);

glRotatef(alpha, 0, 1, 0);

glRotatef(theta, 1, 0, 0);

glColor3f(0.0, 0.0, 0.0);

glutWireTorus(50, 150, 20, 20);

glFlush();

}

int main(int argc, char\* argv[]) {

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB | GLUT\_DEPTH);

glutInitWindowSize(1000, 1000);

glutCreateWindow("Parallel and Perspective Projections");

init();

glutDisplayFunc(display);

glutSpecialFunc(keyPress);

glutMainLoop();

return 0;

}

**OUTPUT:**





